CLAIMS

1. A system for controlling asynchronous updates to a storage location, the system comprising:

a generally accessible register that is asynchronously updateable by hardware and software; and

protection logic in communication with the register, wherein the protection logic includes circuitry to prevent a hardware update to the register from being overwritten by a software update.

2. The system of claim 1 wherein the register includes one or more bits and the protection logic is in communication with one of the bits.

3. The system of claim 1 wherein the register includes a plurality of bits and the protection logic is in communication with one or more of the bits.

4. The system of claim 1 wherein the software update is being performed by a read-modify-write instruction.

5. The system of claim 4 wherein the protection logic:

remembers the value of the bit read by the software in the read portion of the read-modify-write instruction;

compares the value of the bit read by the software to the current value of the bit; and

prevents the software from performing the write portion of the read-modify-write instruction for the bit if the value of the bit read by the software is not the same as the current value of the bit.

6. The system of claim 4 wherein the hardware update occurs after the read portion of the read-modify-write instruction and before the write portion of the read-modify-write instruction.

7. The system of claim 1 wherein the hardware update is a set operation.

8. The system of claim 1 wherein the hardware update is a reset operation.

9. The system of claim 1 wherein the system is a computer processor.

10. The system of claim 9 wherein the computer processor is a server

11. The system of claim 1 wherein the software is millicode.

12. The system of claim 1 wherein the register is a log-trace control register.

13. A method of controlling asynchronous updates to a register, the method comprising:

processing a software read instruction for a bit within the register;

storing the value of the bit in a last read field in response to said processing a software read instruction;

receiving a software write request to the bit;

comparing a current value of the bit to the last read field; and

processing the software write request if the current value of the bit is equal to the last read field.

14. The method of claim 13 wherein the method is implemented in computer hardware.

15. The method of claim 13 wherein the software read instruction and software write request are part of a read-write-modify operation.

16. The method of claim 13 further comprising processing a hardware set operation to the bit after the storing and before the comparing.

17. The method of claim 13 further comprising processing a hardware rest operation to the bit after the storing and before the comparing.

18. The method of claim 13 wherein the software read instruction and software write request are implemented in millicode.

19. A computer program product for controlling asynchronous updates to a register, the computer program product comprising:

a storage medium readable by a processing circuit and storing instructions for execution by the processing circuit for performing a method comprising:

processing a software read instruction for a bit within the register;

storing the value of the bit in a last read field in response to said processing a software read instruction;

receiving a software write request to the bit;

comparing a current value of the bit to the last read field; and

processing the software write request if the current value of the bit is equal to the last read field.